AN

INTERNSHIP REPORT

ON

MEDIA PLAYER MANAGEMENT SYSTEM PROJECT REPORT

BY

KAMAL ACHARYA

(Tribhuvan University)

Date: 2025/06/02

CONTENTS

1. Introduction……...……...3-4
   1.1. Problem Definition…….…….……..…….....3
   1.2. Objectives.…….…….....3
   1.3. Organization of Project Report..……4

2. Review……..…….....5
   2.1. Motivation……..…….…….....5
   2.2. Benefits of the system.…….……..5

3. Requirement Specification……..……6-8
   3.1. Functionality Requirements…….....6
   3.2. Performance Requirements…….…….....6
   3.3. External Interface Requirements.…….……...6

4. Design……..……..9-16
   4.1. Software Engineering Paradigm..……...9
   4.2. DFD Diagrams…….……..10
   4.3. State Transition Diagram.…….…….....16

5. Implementation.…….....17-34
   5.1. System Implementation.…….....17
   5.2. Snapshots……..19

6. Analysis……..35-38
   6.1. System Analysis……...35
   6.2. System Study.……...36
   6.3. System Maintenance……37

7. Testing……..40-44
   7.1. The Testing Process……41

7.2. Testing Strategies……42

8. Conclusion……45

9. References ……46

---

1. Introduction

1.1. Problem Definition

Now a day everyone listens to music and have the need to install a player in their respective devices (PCs, laptops, etc.) to play desired songs and watch videos. Media player is one candidate among all the players which enables a user to listen and play songs/music whenever he/she wants. Media Player allows a normal user to open the media files in which the most popular formats include .mp3

Media Player provides options for play, pause, stop, creation of playlist, etc. Media Player is a small compact player but with great abilities. The entire "Problem Definition" includes the design of this media player so as to provide a nice interface to the users and easiness to listen the music.

Recently, MP3 become popular in this generation. Most of the software companies develop so many types of players which support MP3 file (e.g. Winamp, Window Media Player, Real, RealOne, etc).

Media Player is basically a facility for music lovers. This player has an inbuilt ID3 tag editor. To achieve this we have chosen Java as the implementation language as Java is platform independent. Hence, the application can be run on any platform following Java's "Write once run anywhere" philosophy.

1.2. Objectives

The main purpose of Media Player is to produce an audio (MP3) and video (MPG) player.

The goals of Media Player are:

- Provide a platform to play audio (MP3) file.
- Provide a platform to play video (MPG) file.
- Support playlist (M3U) file.
- Provide playlist management.

- Provide online searching for music and videos.

1.3. Organization of Project Report

The main purpose of this document is to present a detailed description of the functionalities offered by Media Player. It will explain the purpose and features of the system, the interfaces of the system, what system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the developers and users of the system.

2. Review

2.1. Motivation

Presently users use software like Winamp, Windows Media Player and Open Source's VLC Player. Plug-ins for these software are available to control these players.

The good thing with these popular software is, they are rich in features. They have graphic equalizer and support for custom skins, custom plug-ins for extensibility.

2.2. Benefits of the system

Media Player has been developed using Java which offers the following properties:

- Simple
- Secure
- Portable
- Object-oriented
- Robust
- Multithreaded
- Architecture-neutral
- Interpreted
- High Performance
- Distributed
- Dynamic
- Platform- Independency

The Media Player application therefore has the following benefits:

- It is a fast standalone player.
- Facility to save play-list files.
- ID3 tag editor to edit mp3 metadata.
- Made in java, hence is extensible, platform independent, robust.
- Provision of shopping as well searching music online.

3. Requirement Specification

3.1. Functionality Requirements

The following requirements were laid out for our project, Media Player:

- The player can play audio (MP3) file.
- The player can play video (MPG) file.
- The player can provide some necessary functions of playing audio file, such as previous, next, pause, stop, creation of playlist, etc.
- Users can add new songs to the playlist or manage their playlist
- All user's data and playlists could be stored electronically.

3.2. Performance Requirements

Requirements Analysis encompasses those tasks that go into determining the requirements of a new or altered system, taking account of the possibly conflicting requirements of the various stakeholders, such as users. Requirements analysis is critical to the success of a project.

Systematic requirements analysis is also known as requirements engineering. It is sometimes referred to loosely by names such as requirements gathering, requirements capture, or requirements specification. The term "requirements analysis" can also be applied specifically to the analysis proper (as opposed to elicitation or documentation of the requirements, for instance).

3.3. External Interface Requirements

3.3.1. User Interface:

The user will interact with the system through the modules designed in NetBeans IDE which provide a Graphical User Interface (GUI). The user can play audio/ video files, create playlists, manage playlists and search online.

3.3.2. Hardware Interface:

Hardware is the backbone of any up and running software. It becomes more so important in case the applications are on a network or the size of the database and system is very large. Every system is very large. Every system demands a good hardware to support its functions. The hardware requirements are as follows:

a) Processor:

- Minimum: 733 MHz Pentium.
- Recommended: 1 Ghz or higher.

b) Memory:

- Minimum: 256 MB (Application Area) and 128 MB (development purposes).
- Recommended: 512 MB or higher.

c) Disk:

- Minimum: 20 MB
- Recommended: For install it need free space with 30MB.

3.3.3. Software Interface:

This deals with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed. For developing this website we use .net framework. This framework includes an in-built function and it is very user friendly environment. The project has been developed using the following programming environment:

- Java Environment:

  Java environment includes a large number of development tools and hundreds of classes and methods. The development tools are part of the system known as *Java Development Kit (JDK)* and the classes and methods are part of the *Java*

Standard Library (JSL), also known as the *Application Programming Interface (API)*.

- Java Media Framework:

  The Java Media Framework (JMF) is a recent API for Java dealing with real-time multimedia presentation and effects processing. JMF handles time-based media, media which changes with respect to time. Examples of this are video from a television source, audio from a raw-audio format file and animations. The beta JMF 2.0 specification will be used for this report, as they currently reflect the features that will appear in the final version.

- ID3 Tag:

  ID3 is a metadata container most often used in conjunction with the MP3 audio file format. It allows information such as the title, artist, album, track number, or other information about the file to be stored in the file itself. There are two unrelated versions of ID3: ID3v1 and ID3v2. The ID3v1 tag occupies 128 bytes, beginning with the string TAG. To maintain comparability with older media players, the tag was placed at the end of the file. Some players played a small burst of static when they read the tag, but most of them ignored it, and almost all modern players will correctly skip it.

4. Design

4.1. Software Engineering Paradigm

A software engineer or a team of engineers must incorporate a development strategy that encompasses the process, methods, tools and layers. This strategy is often known as the *process model* or a *software engineering paradigm*.

A process model for software engineering is chosen based on the nature of the application, the methods and tools to be used, and the controls and deliverables that are required.

In the development of Media Player, the *Linear Sequential Model* has been applied. This model suggests a systematic, sequential approach to software development that begins at the system level and progresses through analysis, design, coding, testing and supports.

The *Linear Sequential Model* encompasses the following activities:

- System/ Introduction engineering and modeling:

  System engineering and analysis compasses requirement gathering at the system level with small amount of top - level design and analysis.

- Software requirement analysis:

  The requirement gathering process is intensified and focused specifically on software. To understand nature of program, our team has made information domain for the software, as well as required function, behavior, performances and interface. Requirements for both the system and application documented and reviewed with the customer by the team of the team of engineers.

- Design:

  Application design is a multi-step process in which we focused on four distinct attributes of program: Form Properties, software architecture, interface representation, procedure details.

- Code Generation:

  The Design is performed into the detailed manner so that code generation can be accomplished mechanistically.

- Testing:

  After the generation of code program testing begins. The testing process focuses on the logical internals of the application, ensuring that all the statements have been tested fully. Our team conducted test to uncover errors and ensure that defined input will produces actual results that agree with required results.

- Supports:

  The team provides software supports as errors may be encountered after the delivery of the software to the customer as customer requires functional or performances enhancements.

4.2. DFD Diagrams

4.2.1. DFD Level 0:

The top level diagram of the proposed system is depicted in Fig. 4.2.1. It has four external entities: File System, Speakers, Playlist and Control panel. The main process takes three types of information from these entities:

- MP3 files from file system.
- File(s) information from Playlist.
- Control Information from Control Interface.
- Audio data from main process to Speakers is the output.

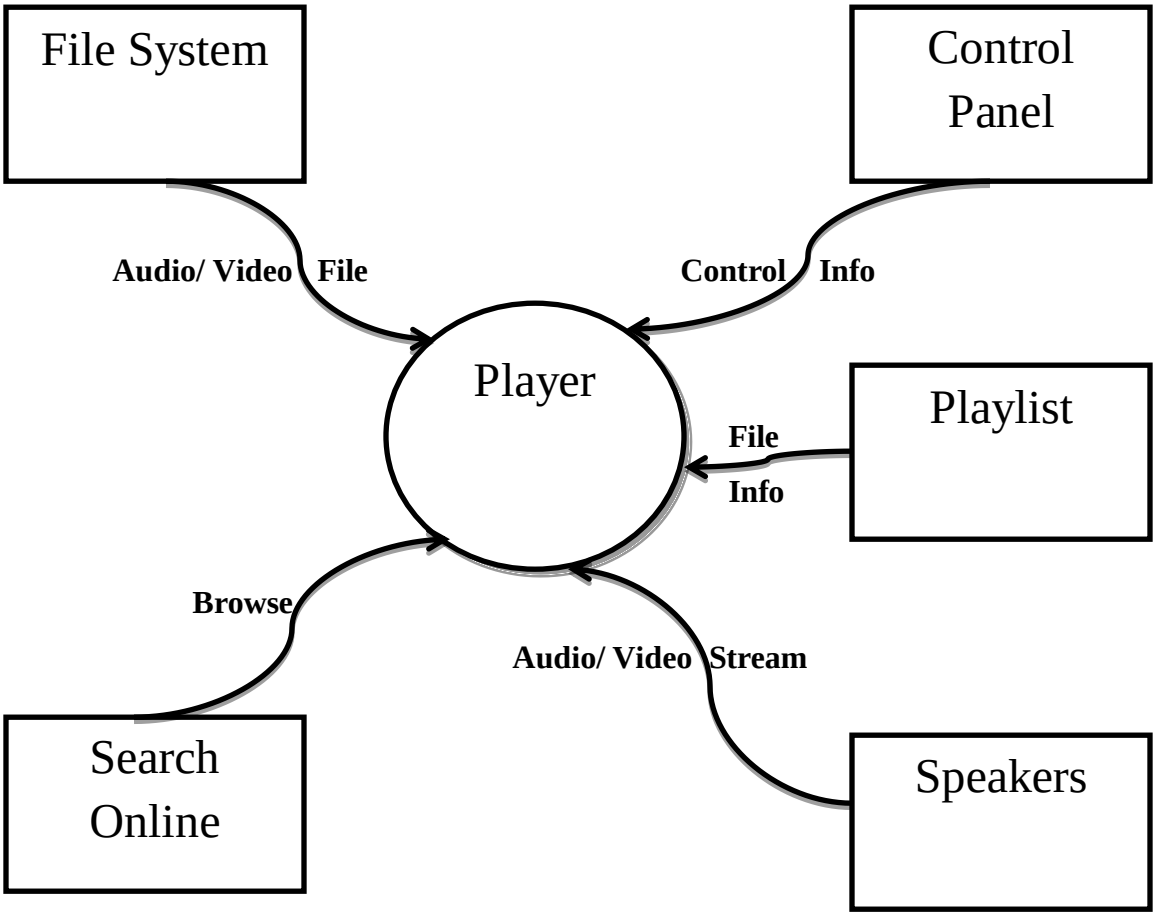

Fig. 4.2.1

4.2.2. DFD Level 1:

The following diagram as depicted in Fig. 4.2.2 sites out three processes:

- Initialize.
- Play.
- Manage Play-List.

The *Initialize* process collects system information and initializes GUI (Graphical User Interface) of our Media Player. It also loads previous settings.

The *Play* process has all the functionalities required for playback like Play, Pause, Stop, Next, Previous, etc.

The *Manage Playlist* process gives facility to Load a saved playlist, Save a playlist, Build a playlist by adding files and folders to it. It also provides an additional facility to edit ID3 header information of MP3 file and save them.

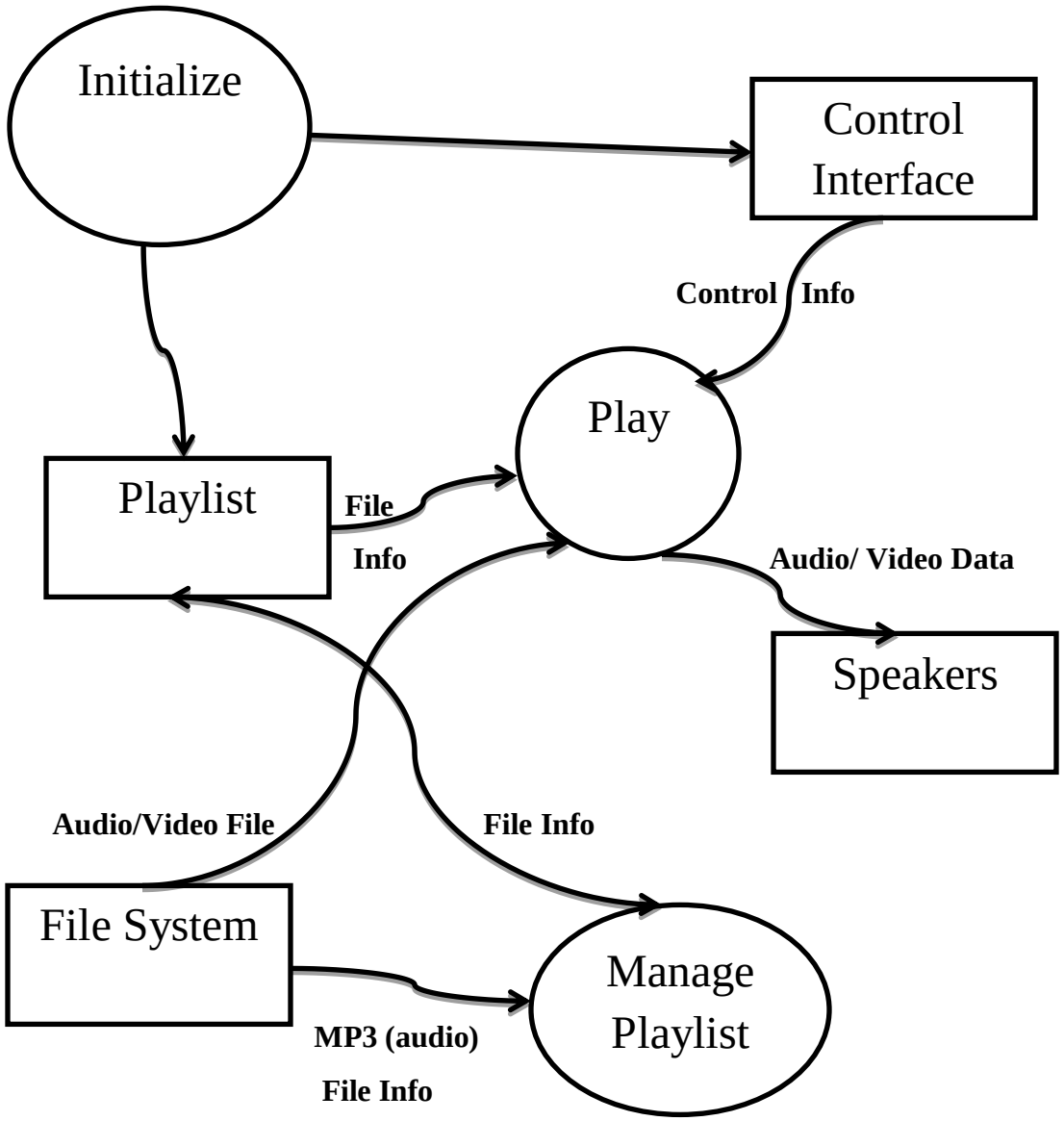
Initialize
Control Interface
Control Info
Play
Playlist
File Info
Audio/ Video Data
Speakers
Audio/Video File
File Info
File System
Manage Playlist
MP3 (audio) File Info

Fig. 4.2.2

4.2.3. DFD Level 2 (Play Process):

The diagram shown in Fig. 4.2.3 depicts the basic functions of Play, Pause, Stop, Next, Previous, Mute, Seek, etc. Only two entities are there in this level. Setting and temporary store which holds the number of current track saves playlist when user quits program.

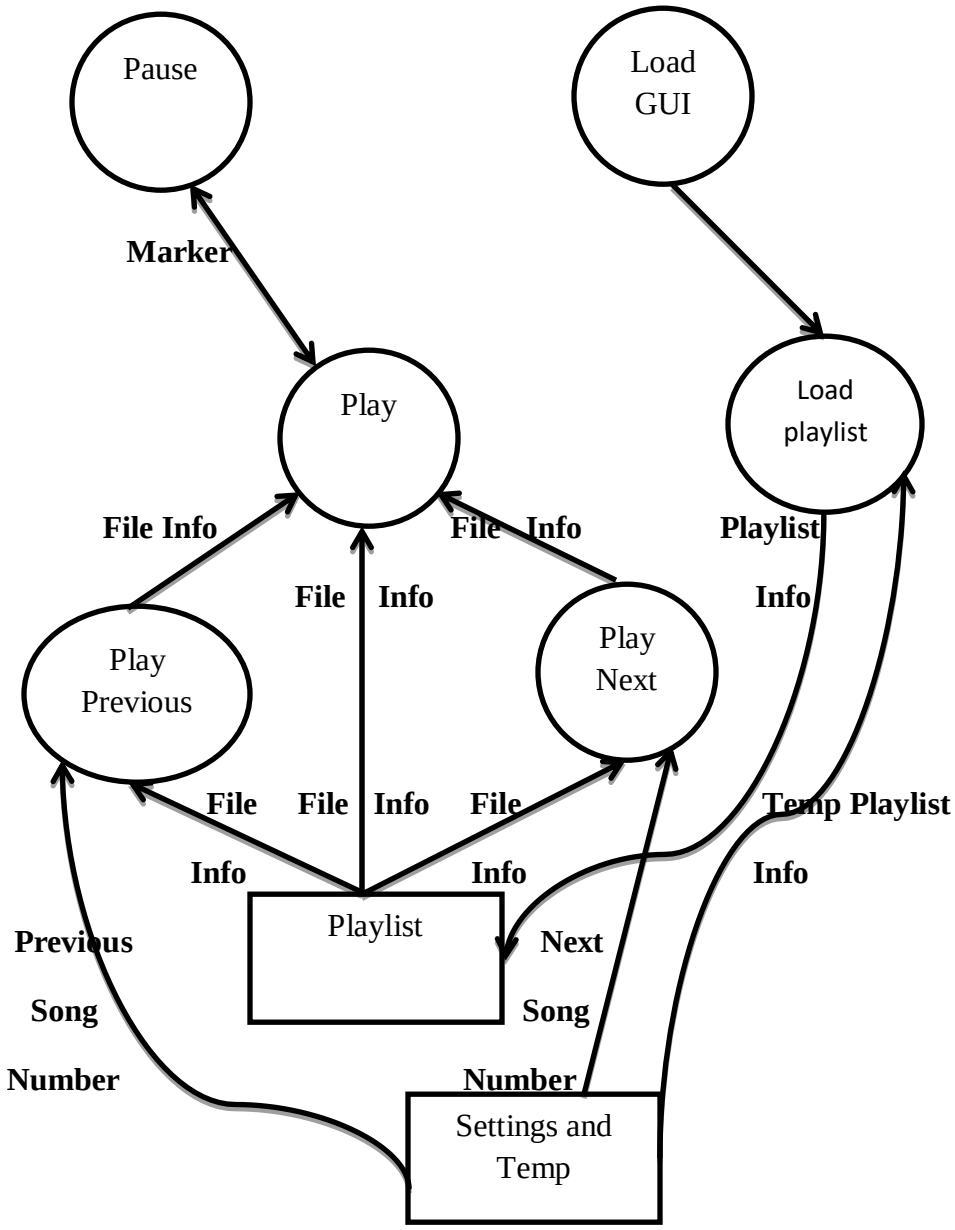

Fig. 4.2.3

4.2.4. DFD Level 2 (Manage Playlist):

The *Manage Playlist* process has been depicted in Fig. 4.2.4. This process performs three tasks:

- Open Playlist.
- Save Playlist.
- Edit ID3 tags of a file.

The *Open* playlist process opens previously saved playlist files of standard type, .m3u.

The *Save* playlist save the currently loaded playlist as .m3u file on the file system.

The *Edit Id3 tag* process is used to edit metadata about mp3 files like album, author, title, duration, year, etc.

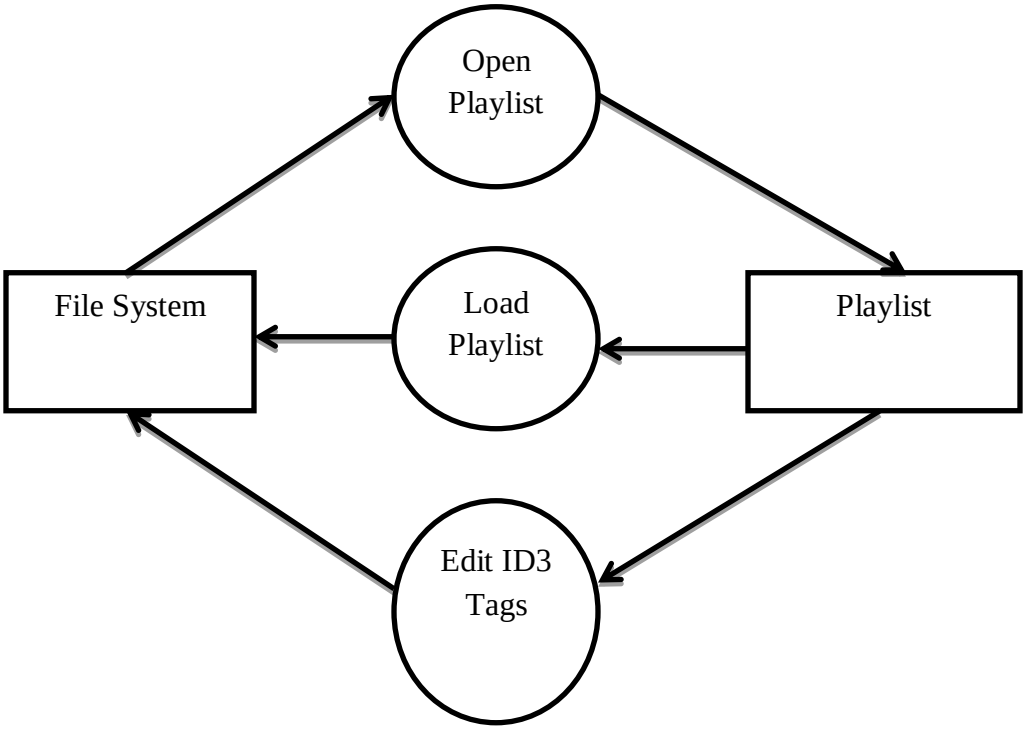

Fig. 4.2.3

4.3. State Transition Diagram

The State Transition Diagram shows various states of the Media Player when user takes different commands. From the diagram shown below in Fig. 4.3, the programmer can easily interpret the required objective and easily understand the functionality.

We have three states in the Media Player:

- Play
- Pause
- Stop

The *Play* state is achieved by taking play action. It can be directed from stop state as well as paused state.

The *Pause* state is achieved when user takes pause command. Commonly there is play-pause state interchange done frequently.

The *Stop* state activates when user takes stop command or playing song finishes.

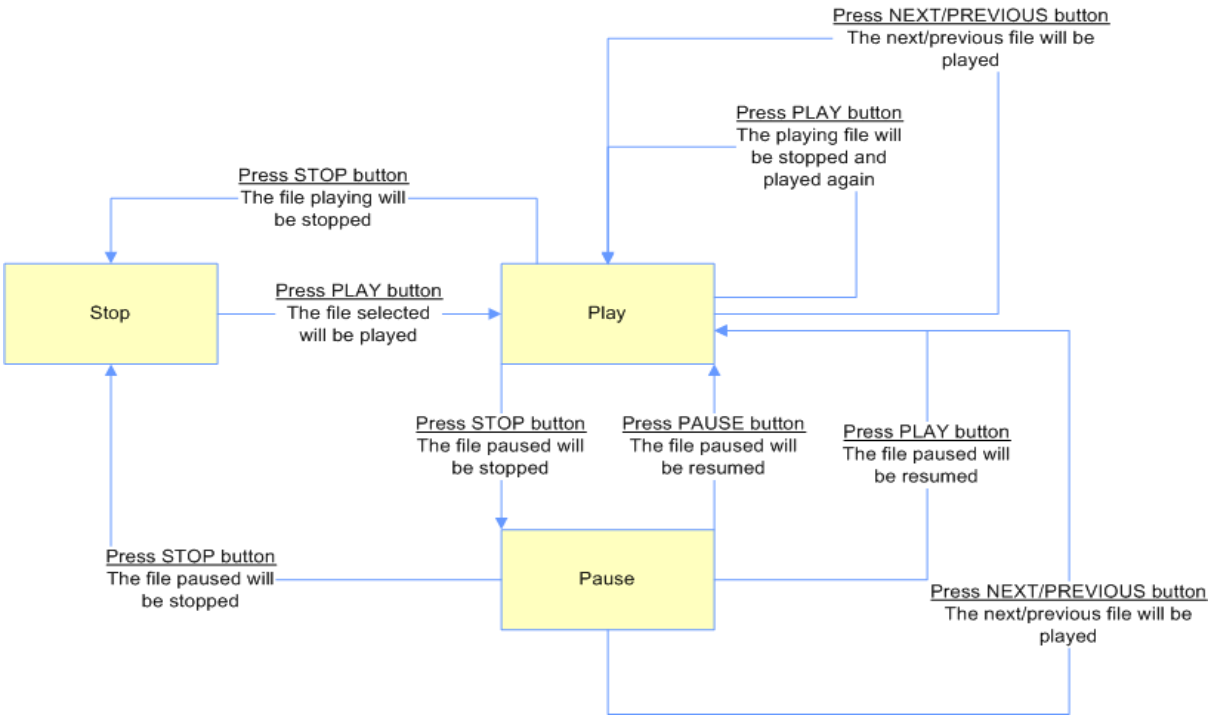
Press NEXT/PREVIOUS button
The next/previous file will be played
Press PLAY button
The playing file will be stopped and played again
Press STOP button
The file playing will be stopped
Stop
Press PLAY button
The file selected will be played
Play
Press STOP button
The file paused will be stopped
Press PAUSE button
The file paused will be resumed
Press PLAY button
The file paused will be resumed
Press STOP button
The file paused will be stopped
Pause
Press NEXT/PREVIOUS button
The next/previous file will be played

Fig. 4.3

5. Implementation

5.1. System Implementation

Implementation is the realization, application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy.

In information technology, an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system. Many implementations may exist for a given specification or standard. For example, web browsers contain implementations of World Wide Web Consortium-recommended specifications, and software development tools contain implementations of programming languages.

After having the user acceptance of the new system developed, the implementation phase begins. Implementation is the stage of a project during which theory is turned into practice. During this phase, all the programs of the system are loaded onto the user's computer. After loading the system, training of the users starts. The main topics of such type of training are:

- How to execute the package.
- How to enter the data.
- How to process the data (processing details).
- How to take our reports.

After the users are trained about the computerized system, manual working has to shift from manual to computerized working. The following two strategies are followed for running the system:

a) Parallel Run:

In such a run for a certain defined period, both the systems, i.e. computerized and manual are executed in parallel. This strategy is helpful because of the following:

- Manual results can be compared with the results of the computerized system.
- Failure of the computerized system at the early stage, does not affect the working of the organization, because the manual system continues to work, as it used to do.

b) Pilot Run:

In this type of run, the new system is installed in parts. Some part of the new system is installed first and executed successfully for considerable time period. When the results are found satisfactory then only other parts are implemented. This strategy builds the confidence and the errors are traced easily.

5.2. Snapshots

5.2.1. Home Page:

Fig. 5.2.1

5.2.2. Play Audio:

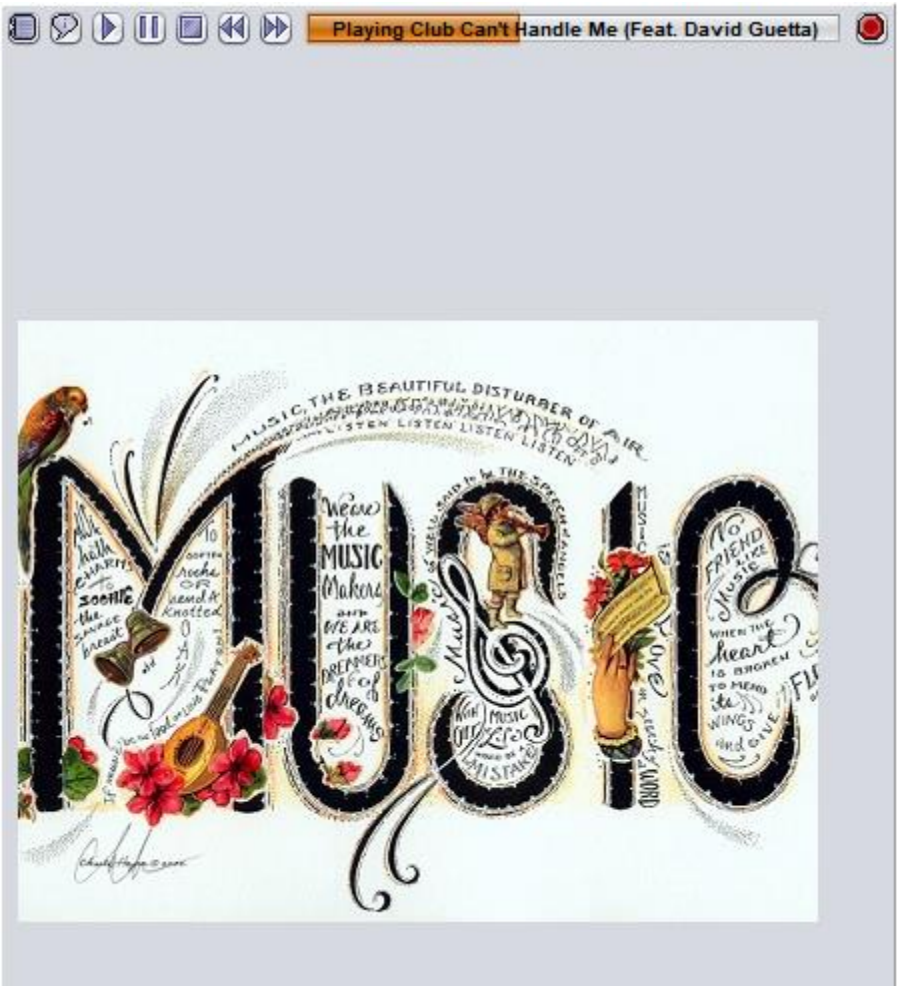
Playing Club Can't Handle Me (Feat. David Guetta)
MUSIC, THE BEAUTIFUL DISTURBER OF THE AIR
LISTEN LISTEN LISTEN LISTEN
We're the MUSIC Makers
and WE ARE the DREAMERS of DREAMS
NO FRIEND LIKE Music
WHEN THE heart IS BROKEN
TO MEND its WINGS and GIVE

Fig. 5.2.2

5.2.3. Create playlist:

Playlist Manager
File
Edit
New
Save
Save As
Save Option
Exit
Track Title
Album
Artist
Time
bloodstream
Stateless
Stateless
5:23
Never Say Never
The Fray
The Fray
4:19
Say All I need
The Vampire Diaries
One Republic
1:12
Here Without You
Away from the Sun
3 Doors Down
3:24
Club Can't Handle Me (Feat. David Guetta)
Step Up 3D (Original Motion Picture Soun...
Flo Rida
3:54
13:25
7 7/15 Beggin' Step Up 3D (Original Motion Picture Soun... Madcon 3:40
9 14 The Catalyst A Thousand Suns Linkin Park 4:42
10 11 Cure For The Itch Hybrid Theory Linkin Park 2:35
10 23 High Voltage Hybrid Theory Linkin Park 3:31
11 Beat It Thriller Michael Jackson 4:19
13 Smooth Criminal Michael Jackson -549:0-18
14 8/17 They Don't Care About Us The Dutch Collection Michael Jackson -857:0-33
15 4:39
16 13 Break the Ice circus Britney spears -960:0-45
17 1 Circus Britney Spears -715:0-50
18 3/10 Gimme More (Official Remix) Gimme More (Mixes)(2007) Britney Spears feat. T.I. 4:36
19 2 Piece Of Me Blackout Britney Spears 5:21
20 04 Shattered Glass Circus Britney Spears -557:0-11
21 0 03 toxid (britneyspears) 03 toxid (britneyspears 3:19
22 1 Nine Thou (Superstars Remix) Need for Speed: Most Wanted Original Soundtrack 5:03
23 4 In A Hood Near You Need for Speed: Most Wanted Original Soundtrack -1017:0...
24 Ludacris -298:0-37
25 8 Fortress Europe Need for Speed Undergrund SoundTracks Asian Dub Foundation 3:53
26 15 Get Low (Remix) Big Mike Presents: The Muthafu Lil' Jon Feat. Busta Rhymes & 4:04

Fig. 5.2.3

5.2.4. Adding Files to Playlist:

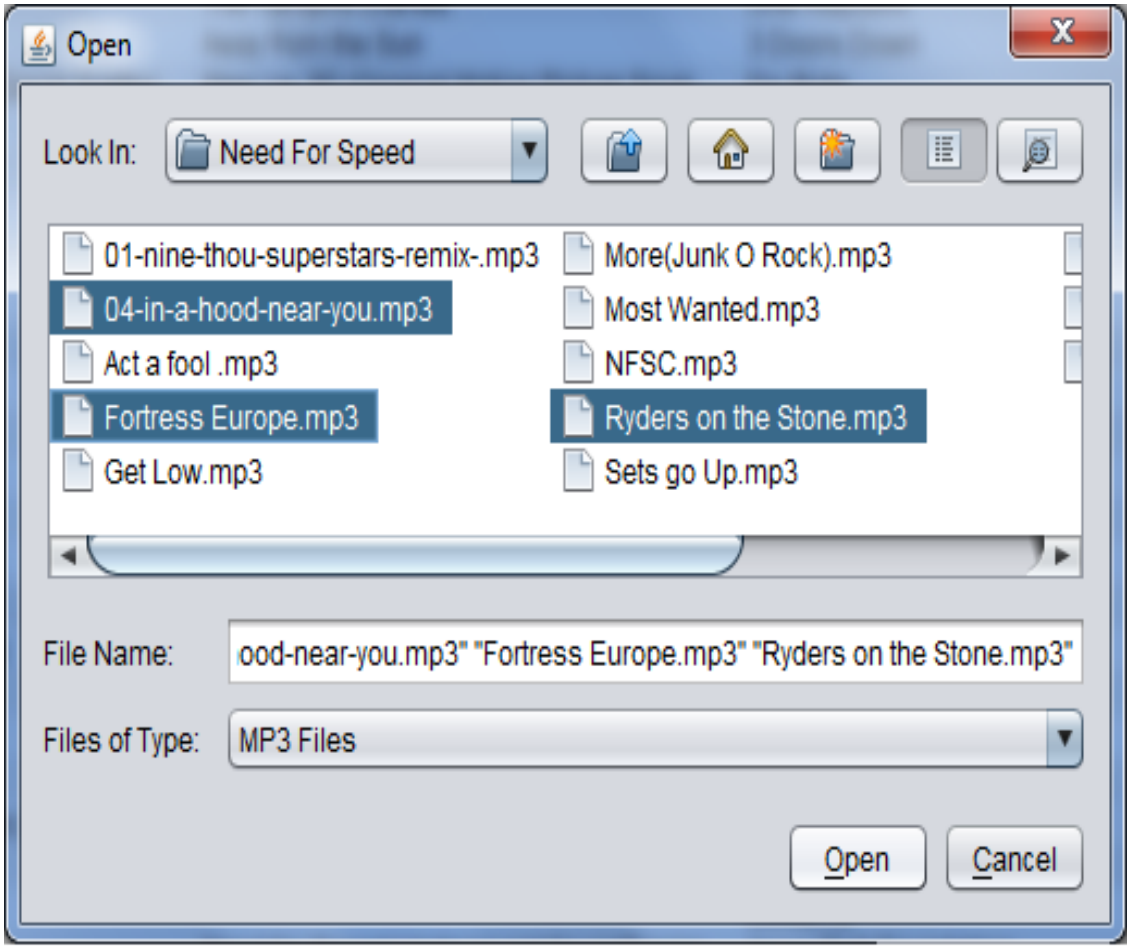

Fig. 5.2.4

5.2.5. Adding Directory to Playlist:

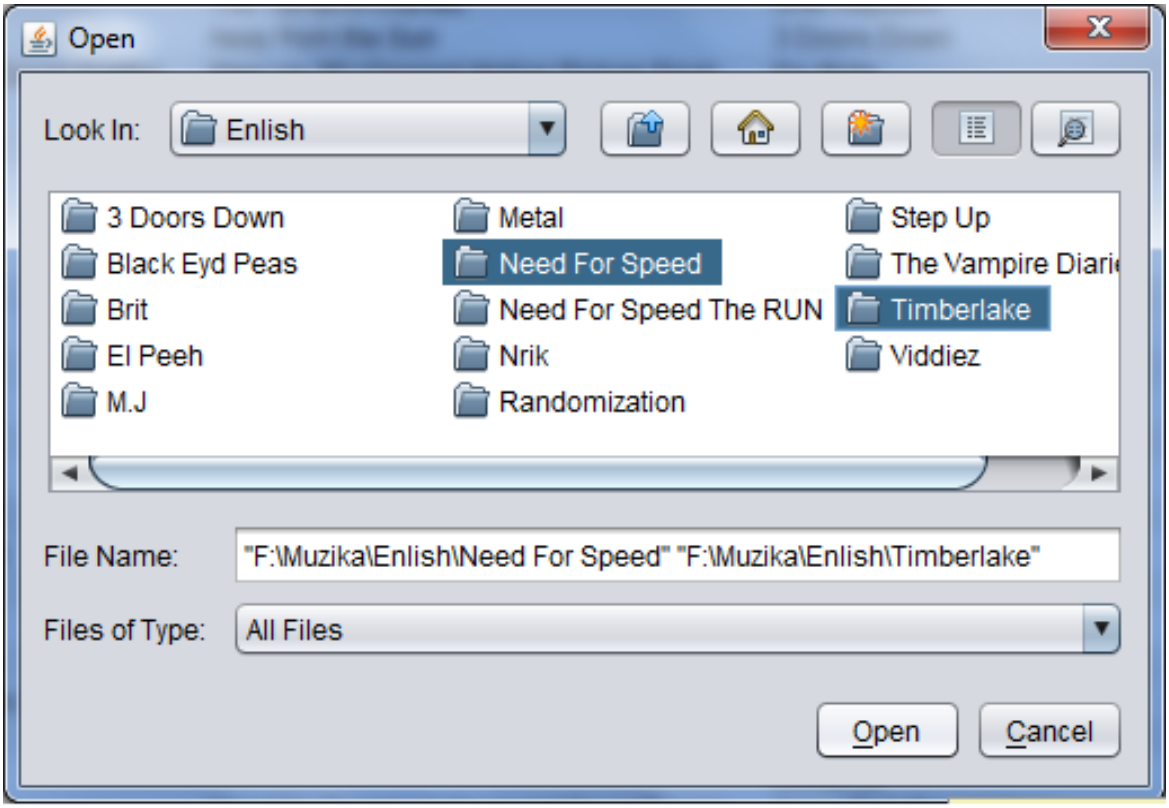

Fig. 5.2.5

5.2.6. Saving a Playlist:

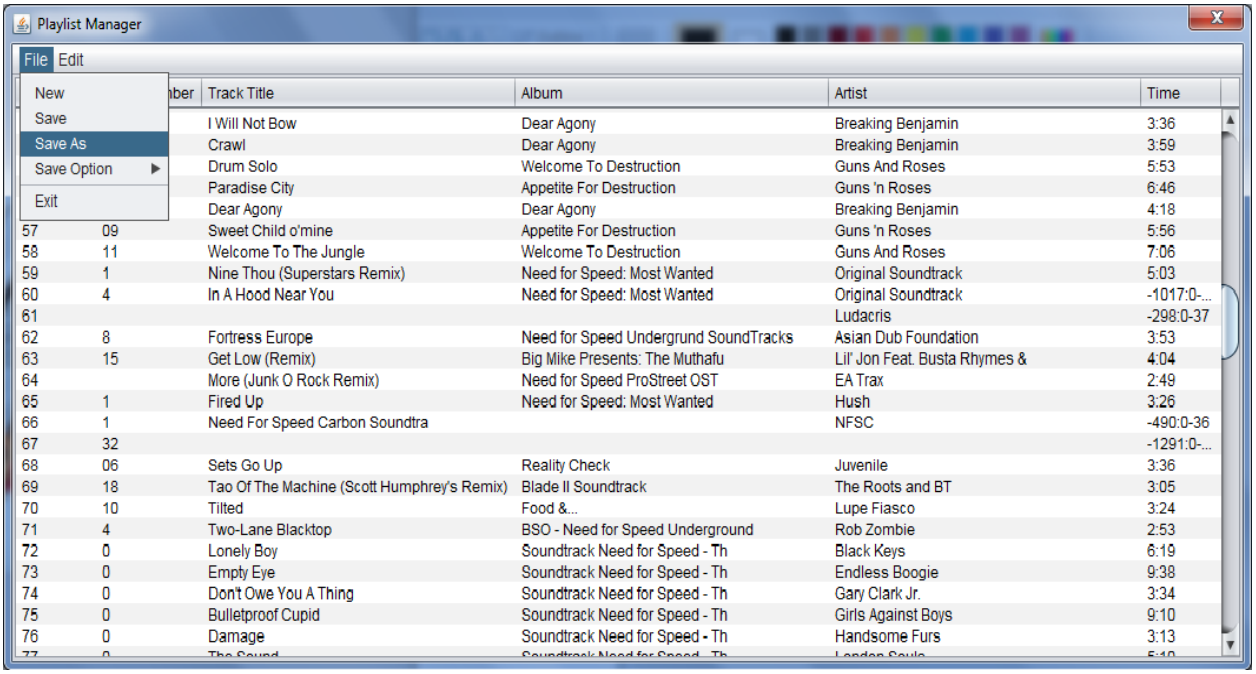
Playlist Manager
File
Edit
New
Save
Save As
Save Option
Exit
Track Title
Album
Artist
Time
I Will Not Bow
Dear Agony
Breaking Benjamin
3:36
Crawl
Dear Agony
Breaking Benjamin
3:59
Drum Solo
Welcome To Destruction
Guns And Roses
5:53
Paradise City
Appetite For Destruction
Guns 'n Roses
6:46
Dear Agony
Dear Agony
Breaking Benjamin
4:18
57 09 Sweet Child o'mine Appetite For Destruction Guns 'n Roses 5:56
58 11 Welcome To The Jungle Welcome To Destruction Guns And Roses 7:06
59 1 Nine Thou (Superstars Remix) Need for Speed: Most Wanted Original Soundtrack 5:03
60 4 In A Hood Near You Need for Speed: Most Wanted Original Soundtrack -1017:0-...
61 Ludacris -298:0-37
62 8 Fortress Europe Need for Speed Undergrund SoundTracks Asian Dub Foundation 3:53
63 15 Get Low (Remix) Big Mike Presents: The Muthafu Lil' Jon Feat. Busta Rhymes & 4:04
64 More (Junk O Rock Remix) Need for Speed ProStreet OST EA Trax 2:49
65 1 Fired Up Need for Speed: Most Wanted Hush 3:26
66 1 Need For Speed Carbon Soundtra NFSC -490:0-36
67 32 -1291:0-...
68 06 Sets Go Up Reality Check Juvenile 3:36
69 18 Tao Of The Machine (Scott Humphrey's Remix) Blade II Soundtrack The Roots and BT 3:05
70 10 Tilted Food &... Lupe Fiasco 3:24
71 4 Two-Lane Blacktop BSO - Need for Speed Underground Rob Zombie 2:53
72 0 Lonely Boy Soundtrack Need for Speed - Th Black Keys 6:19
73 0 Empty Eye Soundtrack Need for Speed - Th Endless Boogie 9:38
74 0 Don't Owe You A Thing Soundtrack Need for Speed - Th Gary Clark Jr. 3:34
75 0 Bulletproof Cupid Soundtrack Need for Speed - Th Girls Against Boys 9:10
76 0 Damage Soundtrack Need for Speed - Th Handsome Furs 3:13

Fig. 5.2.6

5.2.7. Removing an item from Playlist:

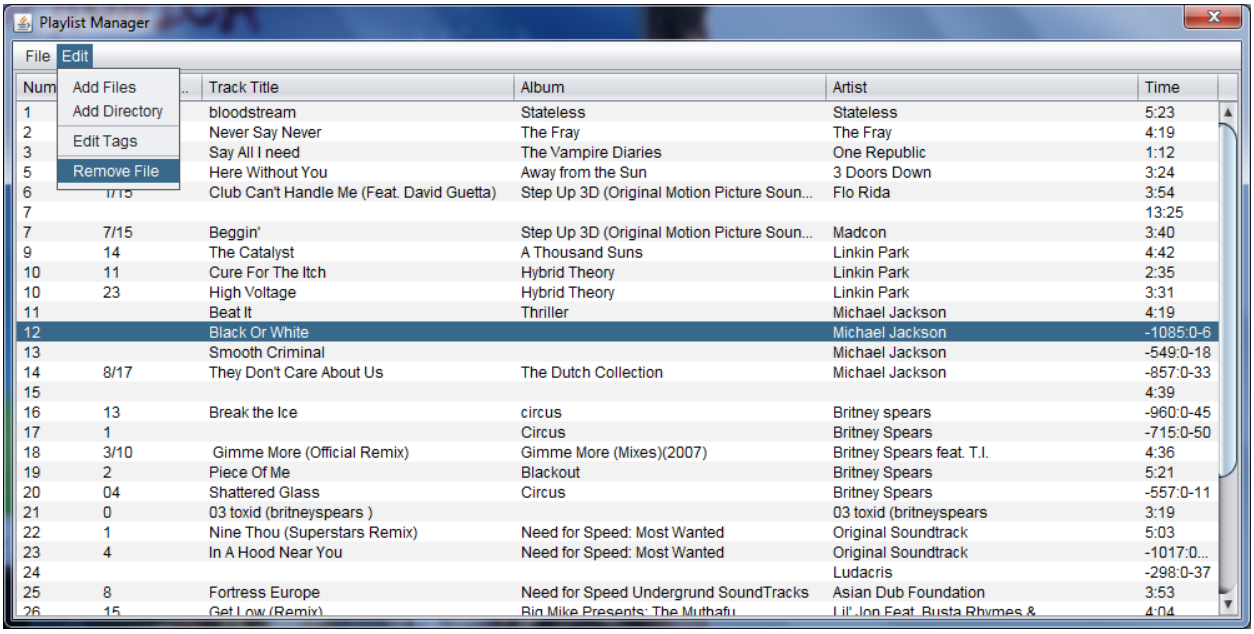
Playlist Manager
File
Edit
Add Files
Add Directory
Edit Tags
Remove File
Num
Track Title
Album
Artist
Time
1 bloodstream Stateless Stateless 5:23
2 Never Say Never The Fray The Fray 4:19
3 Say All I need The Vampire Diaries One Republic 1:12
5 Here Without You Away from the Sun 3 Doors Down 3:24
6 Club Can't Handle Me (Feat. David Guetta) Step Up 3D (Original Motion Picture Soun... Flo Rida 3:54
7 13:25
7 7/15 Beggin' Step Up 3D (Original Motion Picture Soun... Madcon 3:40
9 14 The Catalyst A Thousand Suns Linkin Park 4:42
10 11 Cure For The Itch Hybrid Theory Linkin Park 2:35
10 23 High Voltage Hybrid Theory Linkin Park 3:31
11 Beat It Thriller Michael Jackson 4:19
12 Black Or White Michael Jackson -1085:0-6
13 Smooth Criminal Michael Jackson -549:0-18
14 8/17 They Don't Care About Us The Dutch Collection Michael Jackson -857:0-33
15 4:39
16 13 Break the Ice circus Britney spears -960:0-45
17 1 Circus Britney Spears -715:0-50
18 3/10 Gimme More (Official Remix) Gimme More (Mixes)(2007) Britney Spears feat. T.I. 4:36
19 2 Piece Of Me Blackout Britney Spears 5:21
20 04 Shattered Glass Circus Britney Spears -557:0-11
21 0 03 toxid (britneyspears) 03 toxid (britneyspears 3:19
22 1 Nine Thou (Superstars Remix) Need for Speed: Most Wanted Original Soundtrack 5:03
23 4 In A Hood Near You Need for Speed: Most Wanted Original Soundtrack -1017:0...
24 Ludacris -298:0-37
25 8 Fortress Europe Need for Speed Undergrund SoundTracks Asian Dub Foundation 3:53
26 15 Get Low (Remix) Big Mike Presents: The Muthafu Lil' Jon Feat. Busta Rhymes & 4:04

Fig. 5.2.7

5.2.8. Playing a saved Playlist (.m3u file):

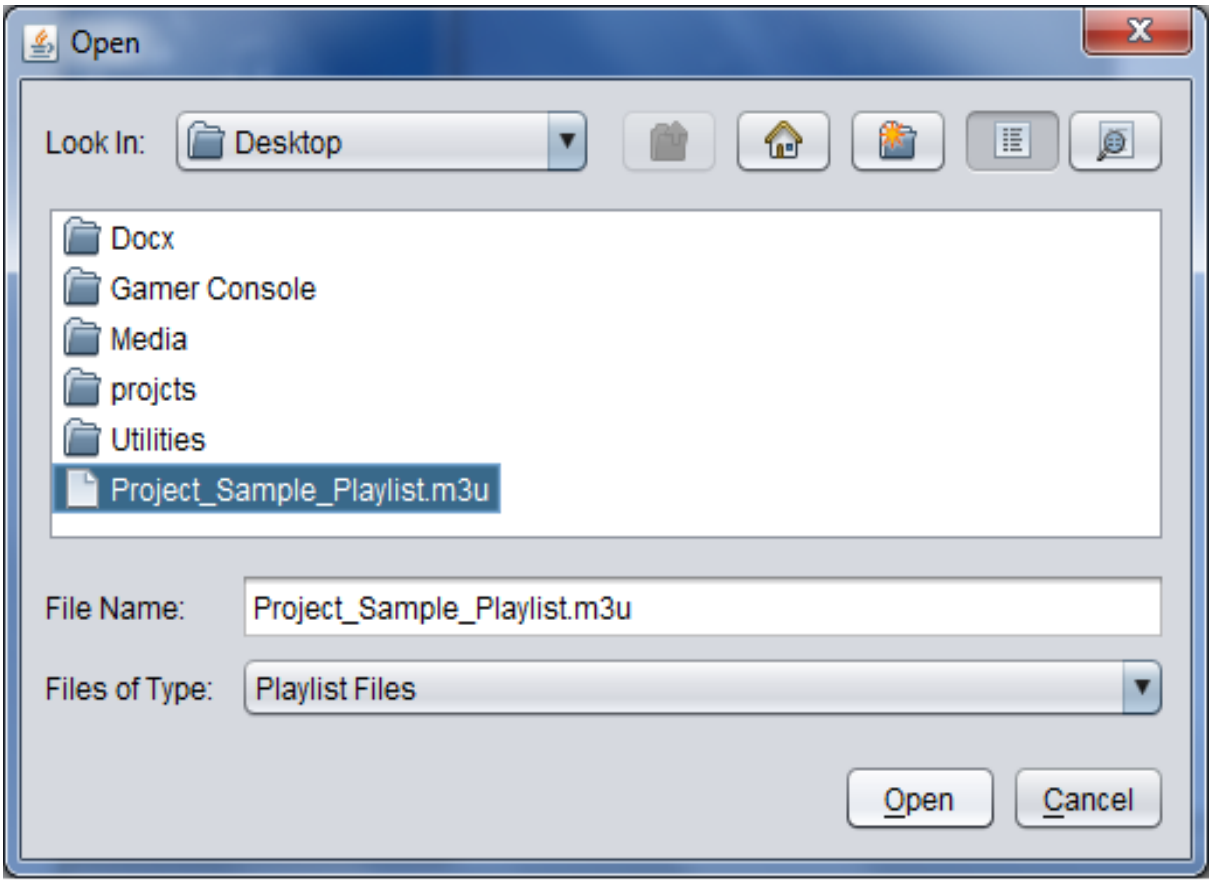

Fig. 5.2.8

5.2.9. Tag Editor:

- Editing Tags in Playlist (Refer to Fig. I)

Playlist Manager
File
Edit
Add Files
Add Directory
Edit Tags
Remove File
Num
Track Title
Album
Artist
Time
1 bloodstream Stateless Stateless 5:23
2 Never Say Never The Fray The Fray 4:19
3 Say All I need The Vampire Diaries One Republic 1:12
5 Here Without You Away from the Sun 3 Doors Down 3:24
6 Club Can't Handle Me (Feat. David Guetta) Step Up 3D (Original Motion Picture Soun... Flo Rida 3:54
7 13:25
7 7/15 Beggin' Step Up 3D (Original Motion Picture Soun... Madcon 3:40
9 14 The Catalyst A Thousand Suns Linkin Park 4:42
10 11 Cure For The Itch Hybrid Theory Linkin Park 2:35
11 High Voltage Hybrid Theory Linkin Park 3:31
11 Beat It Thriller Michael Jackson 4:19
12 Black Or White Michael Jackson -1085:0-6
13 Smooth Criminal Michael Jackson -549:0-18
14 8/17 They Don't Care About Us The Dutch Collection Michael Jackson -857:0-33
15 4:39
16 13 Break the Ice circus Britney spears -960:0-45
17 1 Circus Britney Spears -715:0-50
18 3/10 Gimme More (Official Remix) Gimme More (Mixes)(2007) Britney Spears feat. T.I. 4:36
19 2 Piece Of Me Blackout Britney Spears 5:21
20 04 Shattered Glass Circus Britney Spears -557:0-11
21 0 03 toxid (britneyspears) 03 toxid (britneyspears 3:19
22 1 Nine Thou (Superstars Remix) Need for Speed: Most Wanted Original Soundtrack 5:03
23 4 In A Hood Near You Need for Speed: Most Wanted Original Soundtrack -1017:0...
24 Ludacris -298:0-37
25 8 Fortress Europe Need for Speed Undergrund SoundTracks Asian Dub Foundation 3:53
26 15 Get Low (Remix) Big Mike Presents: The Muthafu Lil'-Jon Feat. Busta Rhymes & 4:04

Fig. I

- Tag Edition (Refer Fig. II)

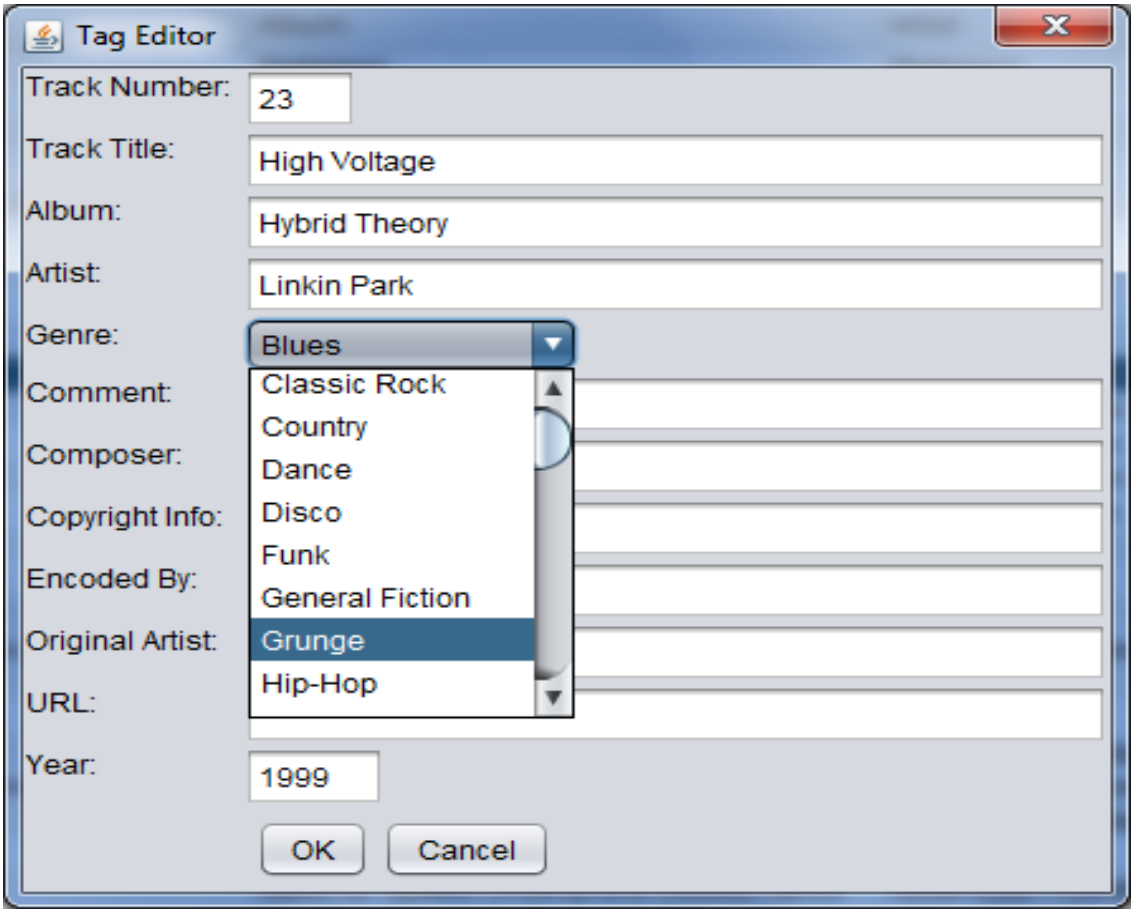

Fig. II

- Final Tag Editor (Refer Fig. III):

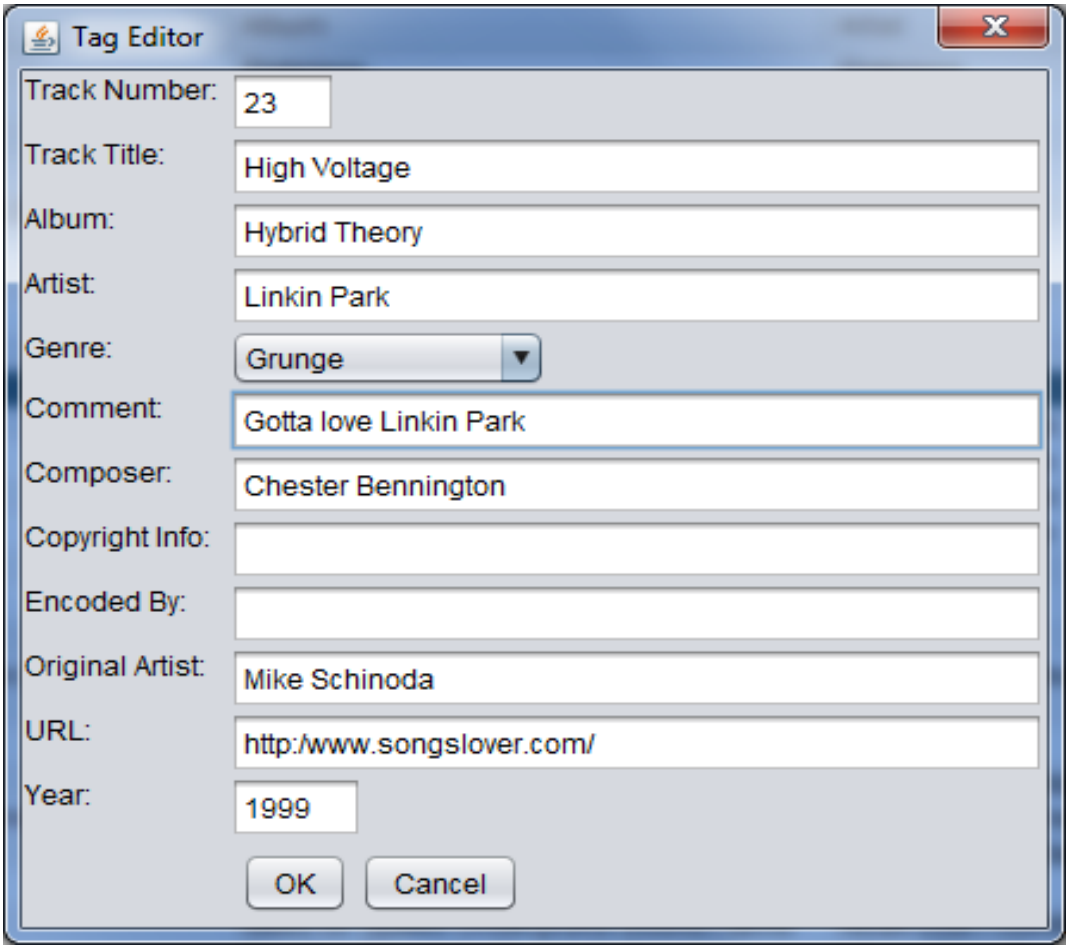

Fig. III

5.2.10. Playing Video:

Fig. 5.2.10

5.2.11. Search Online:

Fig. 5.2.11

5.2.12. Go Google:

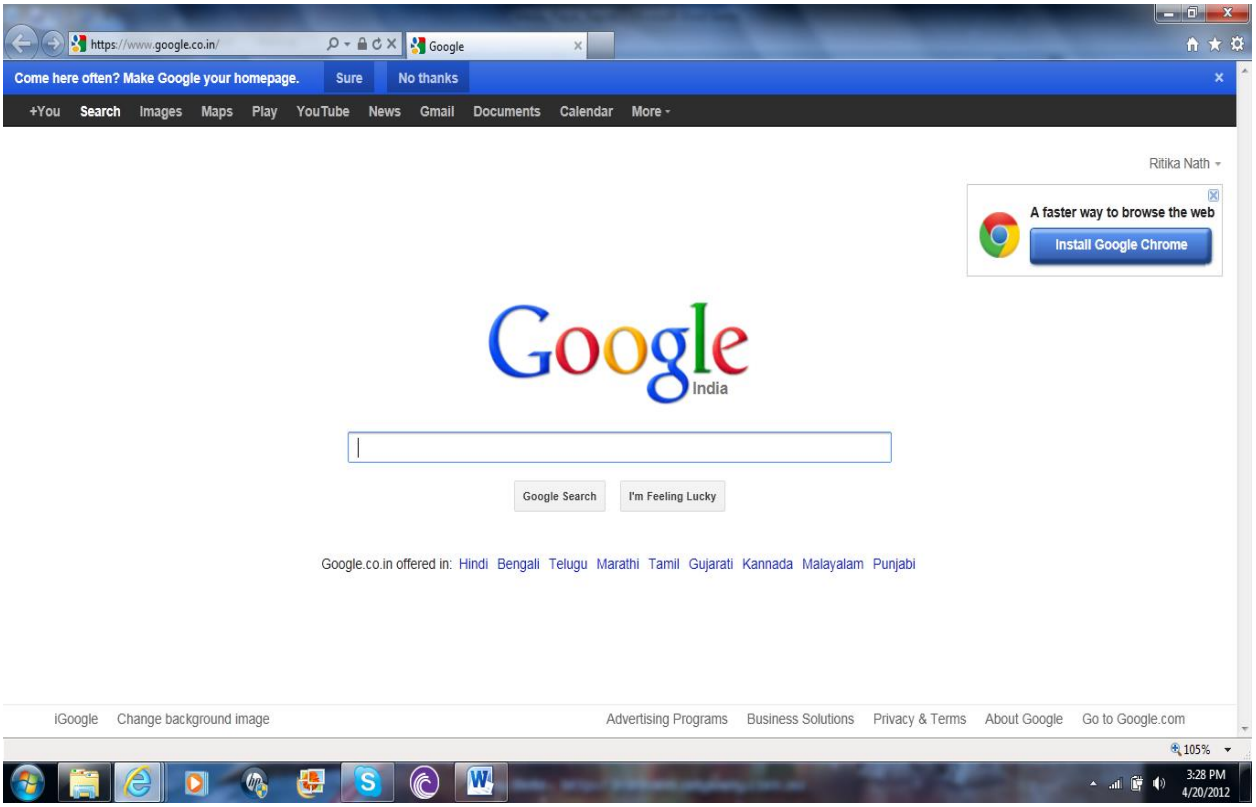

Fig. 5.2.12

5.2.13. About Euphonica Media Player:

ABOUT
Do you love music? Ofcourse you do!!!
Here's something easy to use and install for all
the Music Lovers!!!!
The "Euphonica Media Player" has been
developed on a Java platform.
It provides the following features:
1. Playing audio/ video files.
2. Creation and management of playlists.
3. Addition of the desired directory in the
player and scrolling through and playing
the songs in it.
Hope you enjoy using it!!!!!
Back

Fig. 5.2.13

6. Analysis

6.1. System Analysis

System Analysis by definition is a process of systematic investigation for the purpose of gathering data, interpreting the facts, diagnosing the problem and using this information to either build a completely new system or to recommend the improvements to the existing system.

System analysis is the detailed study of the various operations performed by a system and their relationships within and outside the system. This is the phase of the system development where the user requirements are defined. Present system is studied. Problem is verified and the performance expected by the proposed system is calculated.

The first step in the system development process is the initial investigation. Verification of the user request and is requirements, identification of the problem, determining the feasibility of the request etc., are the major aims to be kept in mind in this phase.

Fact- finding is the first step in the initial investigation. It includes a review of the written documents, on side observations, interviews etc. The next step is the fact analysis, which evaluates the elements of the input and the output of the given system. Data flow evaluates and flow charts are prepared in this phase.

The outcome of the initial investigation is to determine whether the alternative system is feasible in the present circumstance.

Approval of the document, detailing the findings of the initial investigations, initiates the feasibility study, which leads to the selection of the best candidate system.

A satisfactory system analysis involves the process of examining a business situation with the intent of improving it through better methods and procedures. In its core sense, the analysis phase defines the requirements of the system and the problems which user is trying to solve irrespective of how the requirements would be accomplished.

6.2. System Study

This gives a clear picture of what actually the physical system is. The system study is done in two phases:

- In the *first phase*, the preliminary survey of the system is done which helps in identifying the scope of the system.
- The *second phase*_of the system study is more detailed and in-depth study in which the identification of user's requirement and the limitations and problems of the present system are studied.

After completing the system study, a system proposal is prepared by the System Analyst and placed before the user.

On the basis of result of the initial study, feasibility study takes place. The feasibility study is basically the test of the proposed system in the light of its workability, meeting user's requirements, effective use of resources and the cost effectiveness. The main goal of feasibility study is not to solve the problem but to achieve the scope. In the process of feasibility study, the cost and benefits are estimated with greater accuracy.

Based on the user requirements and the detailed analysis of a new system, the new system must be signed. Normally the design phase proceeds in two stages:

- Preliminary or General Design: In the preliminary design, the features of the new system are specified. If the project is still considered to be feasible, we move to the detailed design.

- Structure or Detailed Design: Structure design is a blue print of a computer system solution to a given problem having the same components and interrelationship among the same components as the original problem. In this stage programming language and the platform in which the new system will run are also decided.

After designing the new system, whole system is required to be converted into computer understanding language. In this stage defined procedures are transformed into control specifications by help of a computer language.

6.3. System Maintenance

Maintenance is necessary to eliminate errors in the system during its working life and to tune the system to any variations in in its working environment. It has been that there are always some errors found in the system that must be noted and corrected. It also means the review of the system from time to time.

The figure shown below, Fig. 6.3, depicts the entire system implementation for developing the application.

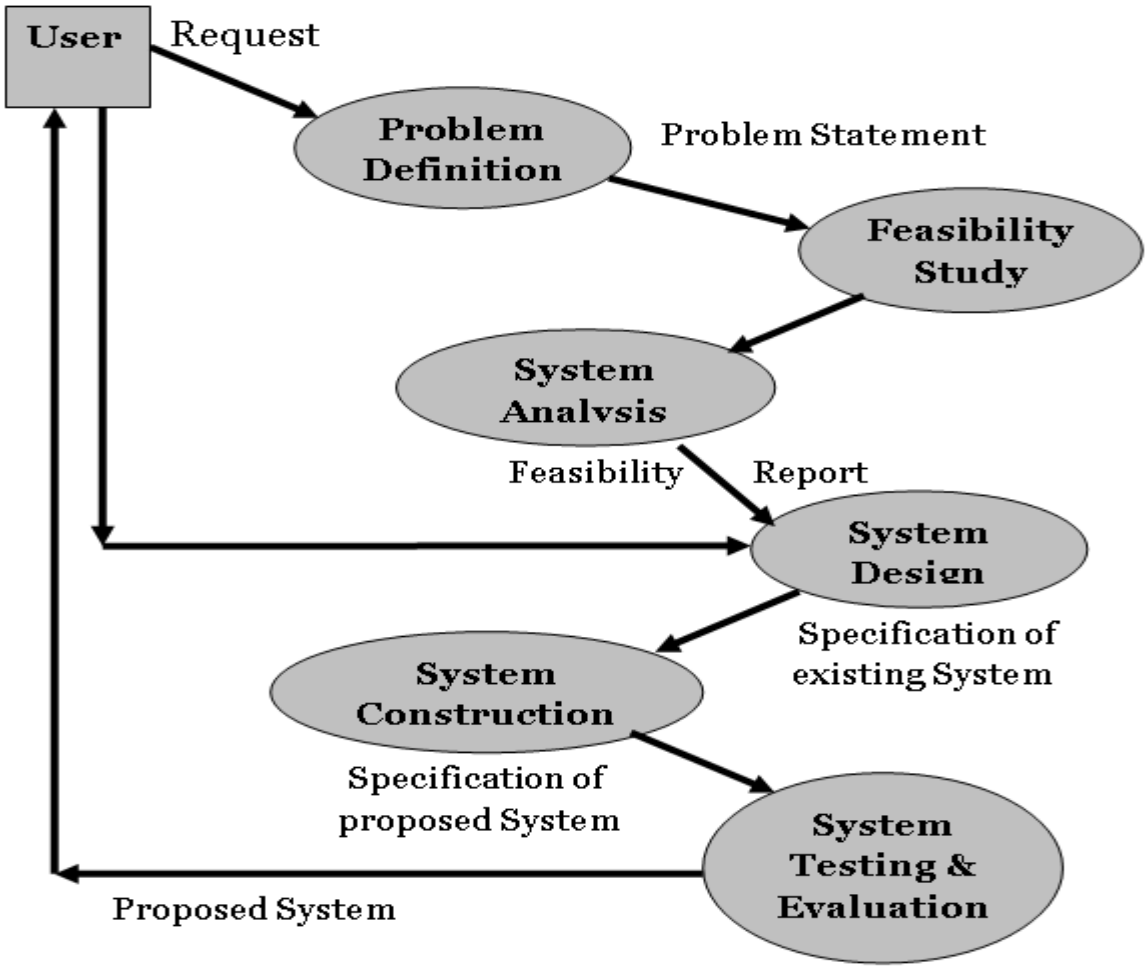

Fig. 6.3

Maintenance includes all the activity after the installation of software that is performed to keep the system operational. Maintenance is necessary to eliminate errors in the system during its working life and to tune the system to any variations in its working environment. It has been seen that there are always some errors found in the system that must be noted and corrected. It also means the review of the system from time to time.

The two major forms of maintenance activities are:

- Corrective Maintenance: It is generally agreed that for large systems, removing all the faults before delivery is extremely difficult and faults will be discovered long after the system is installed. As these faults are detected, they have to be removed. Maintenance activities related to fixing of errors fall under corrective maintenance.

- Adaptive Maintenance: Maintenance also needed due to a change in the environment or the requirements of the system. The introduction of a software system affects the work environment. This change in environment often changes what is desired from the system. Furthermore, often after the system is installed and the users have had a chance to work with it for some time, requirements that are not identified during requirement analysis phase will be uncovered. This occurs, since the experience with the software helps the user to define the needs more precisely. There might also be changes in the input data, the system environment and output formats. All these require modification of the software. The maintenance activities related to such modification fall under adaptive maintenance.

7. Testing

Testing is the major quality control measure employed during software development. Its basic function is to detect errors in the software. During requirement analysis and design, the output is a document that is usually textual and non-executable. After the coding phase, computer programs are available that can be executed for testing phases. This implies that testing not only has to uncover errors introduced during coding, but also errors introduced during the previous phases. Thus, the goal of testing is to uncover requirement, design or coding errors in the programs.

Testing can be defined as a process of executing a program with the aim of finding errors. To perform testing, test cases are designed. A *Test Case* is a particular made up artificial situation upon which a program is exposed so as to find errors. So a good test case is one that finds undiscovered errors. If testing is done properly, it uncovers errors and after fixing those errors we have software that is being developed according to specifications. Test cases are integral part of testing**.**

There are two different approaches to selecting test cases:

- Structural Testing:
  In structural testing the test cases are decided based on the logic of the module to be tested. Structural testing is sometimes called *"Glass box testing"*.

- Functional Testing:
  In functional testing the software for the module to be tested is treated as black box, and then test cases are decided based on the specifications of the system or module. For this reason, this form of testing is also called *"Black box testing"*. The focus is on testing the external behavior of the system. Structural testing is used for lower levels of testing and functional testing is used for higher levels.

7.1. The Testing Process

Testing is an extremely critical and time-consuming activity. It requires proper planning of the overall testing process. Frequently the testing process starts with the *Test plan* (Test configuration). This plan identifies all the testing related activities that must be performed and specifies the schedule, allocates the resources, and specify guidelines for testing. The test plan specifies manner in which the modules will integrate together. Then for different test units, a *Test case specification document* is produced, which lists all the different test cases, together with the expected outputs, that will be used for testing. During the testing of the unit, the specified test cases are executed and actual result is compared with the expected output. The final output of the testing phases is to the *Text report and the Error report*, or set of such reports (one of each unit is tested). Each test report contains the set of such test cases and the result of executing the code with these test cases. The error report describes the errors encountered and action taken to remove those errors. The figure below, Fig. 7.1 shows the testing process.

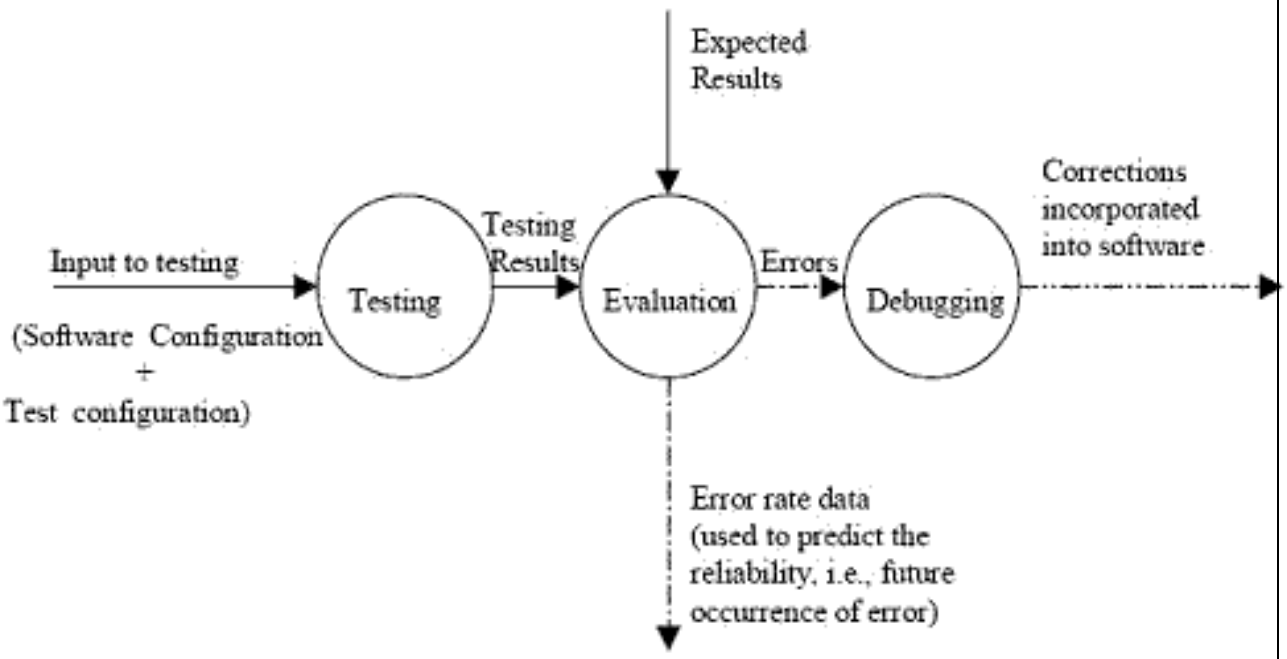

Fig. 7.1

7.2. Testing Strategies

Various software-testing strategies have been proposed so far. Things that are common and important in these strategies are that testing begins at the module level and works "outward": tests which are carried out are done at the module level where major functionality is tested and then it works toward the integration of the entire system. Different testing techniques are appropriate at different points in time. Under different circumstances, different testing methodologies are to be used which will be the decisive factor for software robustness and scalability. Circumstance essentially mean the level at which the testing is being done. The figure below, Fig. 7.2, shows a sequence of tests.

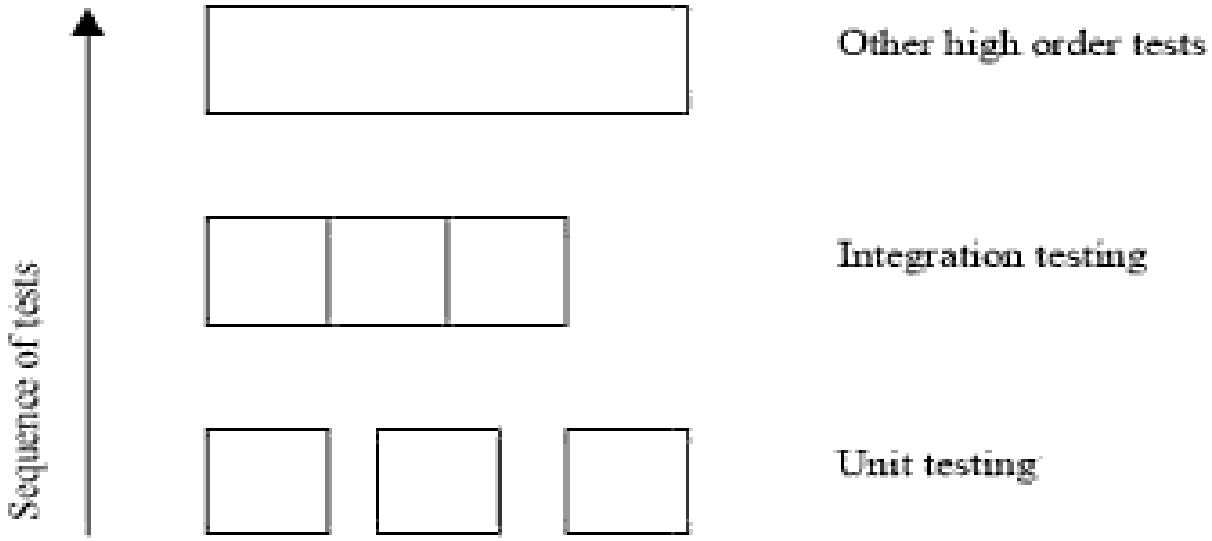

Fig. 7.2

The different testing strategies are:

- Unit Testing:

  The starting po0int of testing is unit testing. The smallest unit of software design is a module. Unit testing is performed to check the functionality of these units. It is done before these modules are integrated together to build the overall system. Since the modules are small in size, individual programmers can do unit testing on their respective modules. So unit testing is basically white box oriented. Procedural design descriptions are used and control paths are tested to uncover errors within individual modules. Unit testing can be done for more than one module at a time.

- Integration Testing:

  Unit testing ensures that all modules have been tested and each of them works properly individually. Unit testing does not guarantee if these modules will work fine if these are integrated together as a whole system. It is observed that many errors crop up when the modules are joined together. The goal of this testing is to detect design errors, while focusing on testing the interconnection between modules.

  There are two approaches in integration testing. One is *Top down integration* and the other is *Bottom up integration.*

- System Testing:

  After the system is put together, system testing is performed. Software is only one element of a larger computer-based system. Ultimately, software is incorporated with other system elements and a series of system integration and validation tests are conducted.

  System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated and perform allocated functions.

The different types of System Tests are:

- Recovery Testing:

  Recovery testing is a system test that forces the software to fail in a variety of ways and verifies that recovery is properly performed.

- Security Testing:

  Security testing attempts to verify that protection mechanism built into a system will protect it from unauthorized penetration.

- User Acceptance Testing:

  In this type of testing, the software is handed over to the user in order to find out if the software meets the user expectations and works as it is expected to. In software development, user acceptance testing (UAT) - also called beta testing, application testing, and end user testing - is a phase of software development in which the software is tested in the "real world" by the intended audience.

  The experiences of the users are forwarded back to the developers who make final changes before releasing the software commercially.

8. Conclusion

In conclusion we believe that the use of *Euphonica Media Player* will ease all the predicaments of the users, providing a robust and platform-independent application for all the music lovers. It will provide functionalities for playing audio/ video files and will also help users to manage their playlists efficiently. ID3 tag editor has been used which is unique in nature as a very few popular media players facilitate for the same.

It also helps, to a great extent, in providing a user- friendly interface and a design which is easy to use. The users will be able to save a lot of time by searching for music online.

Since, the player has been developed using Java, it is robust, platform- independent, simple, fast and reliable, which are the most important requirements in this technical era.

References

1.Kamal Acharya. School management system project report. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254873.34023165/v1

2. Kamal Acharya. A CASE STUDY OF CINEMA MANAGEMENT SYSTEM PROJECT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254873.30191075/v1

3. Kamal Acharya. A CASE STUDY ON ONLINE TICKET BOOKING SYSTEM PROJECT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254872.26972790/v1

4. Kamal Acharya. Web chatting application project report management system. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254871.18588592/v1

5. Kamal Acharya. RETAIL STORE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254871.14590154/v1

6. Kamal Acharya. SUPERMARKET MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252491.19145062/v1

7. Kamal Acharya. SOCIAL MEDIA MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252491.11210579/v1

8. Kamal Acharya. Online music portal management system project report. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252488.89734698/v1

9. Kamal Acharya. COLLEGE BUS MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172245277.70798942/v1

10. Kamal Acharya. AUTOMOBILE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172245276.67982593/v1

11. Kamal Acharya. Ludo management system project report. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172243999.98091616/v1

12. Kamal Acharya. Literature online quiz system project report. Authorea. July 31, 2024 DOI: https://doi.org/10.22541/au.172243825.53562953/v1

13. Kamal Acharya. Avoid waste management system project. Authorea. July 29, 2024. DOI: https://doi.org/10.22541/au.172228528.85022205/v1

14. Kamal Acharya. CHAT APPLICATION THROUGH CLIENT SERVER MANAGEMENT SYSTEM PROJECT. Authorea. July 29, 2024. DOI: https://doi.org/10.22541/au.172228527.74316529/v1

15. Acharya, Kamal, Online Job Portal Management System (May 5, 2024). Available at SSRN: https://ssrn.com/abstract=4817534 or http://dx.doi.org/10.2139/ssrn.4817534

16. Acharya, Kamal, Employee leave management system. (May 7, 2024). Available at SSRN: https://ssrn.com/abstract=4819626 or http://dx.doi.org/10.2139/ssrn.4819626

17. *Acharya, Kamal, Online electricity billing project report. (May 7, 2024). Available at SSRN: https://ssrn.com/abstract=4819630 or http://dx.doi.org/10.2139/ssrn.4819630*

18. *Acharya, Kamal, POLICY MANAGEMENT SYSTEM PROJECT REPORT. (December 10, 2023). Available at SSRN: https://ssrn.com/abstract=4831694 or http://dx.doi.org/10.2139/ssrn.4831694*

19. *Acharya, Kamal, Software testing for project report. (May 16, 2023). Available at SSRN: https://ssrn.com/abstract=4831028 or http://dx.doi.org/10.2139/ssrn.4831028*

20. *Acharya, Kamal, ONLINE CRIME REPORTING SYSTEM PROJECT. (August 10, 2022). Available at SSRN: https://ssrn.com/abstract=4831015 or http://dx.doi.org/10.2139/ssrn.4831015*

21. *Acharya, Kamal, Burger ordering system project report. (October 10, 2022). Available at SSRN: https://ssrn.com/abstract=4832704 or http://dx.doi.org/10.2139/ssrn.4832704*

23. *Acharya, Kamal, Teachers Record Management System Project Report (December 10, 2023). Available at SSRN: https://ssrn.com/abstract=4833821 or http://dx.doi.org/10.2139/ssrn.4833821*

23. *Acharya, Kamal, Dairy Management System Project Report (December 20, 2020). Available at SSRN: https://ssrn.com/abstract=4835231 or http://dx.doi.org/10.2139/ssrn.4835231*

24. *Acharya, Kamal, Electrical Shop Management System Project (December 10, 2019). Available at SSRN: https://ssrn.com/abstract=4835238 or http://dx.doi.org/10.2139/ssrn.4835238*

25. *Acharya, Kamal, Online book store management system project report. (Febuary 10, 2020). Available at SSRN: https://ssrn.com/abstract=4835277 or http://dx.doi.org/10.2139/ssrn.4835277*

26. *Acharya, Kamal, Paint shop management system project report. (January 10, 2019). Available at SSRN: https://ssrn.com/abstract=4835441 or http://dx.doi.org/10.2139/ssrn.4835441*

27. *Acharya, Kamal, Supermarket billing system project report. (August 10, 2021). Available at SSRN: https://ssrn.com/abstract=4835474 or http://dx.doi.org/10.2139/ssrn.4835474*

28. *Acharya, Kamal, Online taxi booking system project report. (March 10, 2022). Available at SSRN: https://ssrn.com/abstract=4837729 or http://dx.doi.org/10.2139/ssrn.4837729*

29. *Acharya, Kamal, Online car servicing system project report. (March 10, 2023). Available at SSRN: https://ssrn.com/abstract=4837832 or http://dx.doi.org/10.2139/ssrn.4837832*

30. *Acharya, Kamal, School management system project report. (July 10, 2021). Available at SSRN: https://ssrn.com/abstract=4837837 or http://dx.doi.org/10.2139/ssrn.4837837*

31. *Acharya, Kamal, Furniture Showroom Management System Project Report (March 21, 2021). Available at SSRN: https://ssrn.com/abstract=4839422 or http://dx.doi.org/10.2139/ssrn.4839422*

32. *Acharya, Kamal, Online Vehicle Rental System Project Report (March 21, 2019). Available at SSRN: https://ssrn.com/abstract=4839429* or *http://dx.doi.org/10.2139/ssrn.4839429*

33. *Acharya, Kamal, Fruit Shop Management System Project Report (August 10, 2023). Available at SSRN: https://ssrn.com/abstract=4841048* or *http://dx.doi.org/10.2139/ssrn.4841048*

34. *Acharya, Kamal, Hall Booking Management System Project Report (December 21, 2023). Available at SSRN: https://ssrn.com/abstract=4841055* or *http://dx.doi.org/10.2139/ssrn.4841055*

35. *Acharya, Kamal, Lundry Management System Project Report (October 21, 2023). Available at SSRN: https://ssrn.com/abstract=4841059* or *http://dx.doi.org/10.2139/ssrn.4841059*

36. *Acharya, Kamal, A CASE STUDY OF CINEMA MANAGEMENT SYSTEM PROJECT (September 25, 2023). Available at SSRN: https://ssrn.com/abstract=4841209* or *http://dx.doi.org/10.2139/ssrn.4841209*

37. *Acharya, Kamal, A CASE STUDY ON ONLINE TICKET BOOKING SYSTEM PROJECT (May 25, 2024). Available at SSRN: https://ssrn.com/abstract=4841210* or *http://dx.doi.org/10.2139/ssrn.4841210*

38. *Acharya, Kamal, ONLINE DATING MANAGEMENT SYSTEM PROJECT REPORT. (April 25, 2023). Available at SSRN: https://ssrn.com/abstract=4842066 or http://dx.doi.org/10.2139/ssrn.4842066*

39. *Acharya, Kamal, ONLINE RESUME BUILDER MANAGEMENT SYSTEM PROJECT REPORT. (April 25, 2021). Available at SSRN: https://ssrn.com/abstract=4842071 or http://dx.doi.org/10.2139/ssrn.4842071*

40. *Acharya, Kamal, TOLL TEX MANAGEMENT SYSTEM PROJECT REPORT (August 21, 2023). Available at SSRN: https://ssrn.com/abstract=4842082 or http://dx.doi.org/10.2139/ssrn.4842082*

41. *Acharya, Kamal, Chat Application Through Client Server Management System Project Report (June 25, 2023). Available at SSRN: https://ssrn.com/abstract=4842761 or http://dx.doi.org/10.2139/ssrn.4842761*

42. *Acharya, Kamal, Web Chatting Application Management System Project Report (April 25, 2022). Available at SSRN: https://ssrn.com/abstract=4842771 or http://dx.doi.org/10.2139/ssrn.4842771*

43. Acharya, Kamal, Automobile management system project report (May 25, 2022). Available at SSRN: https://ssrn.com/abstract=4846917 or http://dx.doi.org/10.2139/ssrn.4846917

44. Acharya, Kamal, College bus management system project report (April 25, 2023). Available at SSRN: https://ssrn.com/abstract=4846920 or http://dx.doi.org/10.2139/ssrn.4846920

45. Acharya, Kamal, Courier management system project report (May 25, 2023). Available at SSRN: https://ssrn.com/abstract=4846922 or http://dx.doi.org/10.2139/ssrn.4846922

46. Acharya, Kamal, Event management system project report (April 25, 2021). Available at SSRN: https://ssrn.com/abstract=4846927 or http://dx.doi.org/10.2139/ssrn.4846927

47. Acharya, Kamal, Library management system project report II (May 25, 2020). Available at SSRN: https://ssrn.com/abstract=4848857 or http://dx.doi.org/10.2139/ssrn.4848857